AAAI-26 / IAAI-26 / EAAI-26
JANUARY 20–27, 2026 | SINGAPORE

Heterogeneous Graph Neural Networks for Assumption-based Argumentation

Preesha Gehlot, Anna Rapberger, **Fabrizio Russo** and Francesca Toni

IMPERIAL

European Research Council
Established by the European Commission
J.P.Morgan
Royal Academy of Engineering

Assumption Based Argumentation Framework

Assumptions | Claims | Rules | Contraries

Example

Assumptions

$\{a, b, c, d\}$

Rules

$\{\mathbf{R1}: \overline{c} \leftarrow a, d \quad \mathbf{R2}: p \leftarrow b \quad \mathbf{R3}: \overline{d} \leftarrow c \quad \mathbf{R4}: \overline{a} \leftarrow p, c\}$

Claims

$\{p, \overline{a}, \overline{b}, \overline{c}, \overline{d}\}$

Contraries

$\{\overline{a}, \overline{b}, \overline{c}, \overline{d}\}$

Resolving Debates

Stable semantics: set of assumptions that does not attack itself and attacks all assumptions outside the set.

ABAF D:

Assumptions = $\{a, b, c, d\}$

Claims = $\left\{p, \overline{a}, \overline{b}, \overline{c}, \overline{d}\right\}$

Rules:

R1: $\overline{c} \leftarrow a, d$

R2: $\mathrm{p} \leftarrow \mathrm{b}$

R3: $\overline{\mathrm{d}} \leftarrow \mathrm{c}$

R4: $\overline{\mathrm{a}} \leftarrow \mathrm{p}, \mathrm{c}$

Contraries

$\left\{\overline{a}, \overline{b}, \overline{c}, \overline{d}\right\}$

$\{a, b, d\}$ **attacks** $\{c\}$ **(R1)**

$\{b, c\}$ **attacks** $\{a\}$ **(R4 and R2)**

$\{b, c\}$ **attacks** $\{d\}$ **(R3)**

stable extensions:

$\{a, b, d\}$

$\{b, c\}$

Motivation

Structured Reasoning with ABA

$\{b, c\}$ attacks $\{a\}$ (R4 and R2)

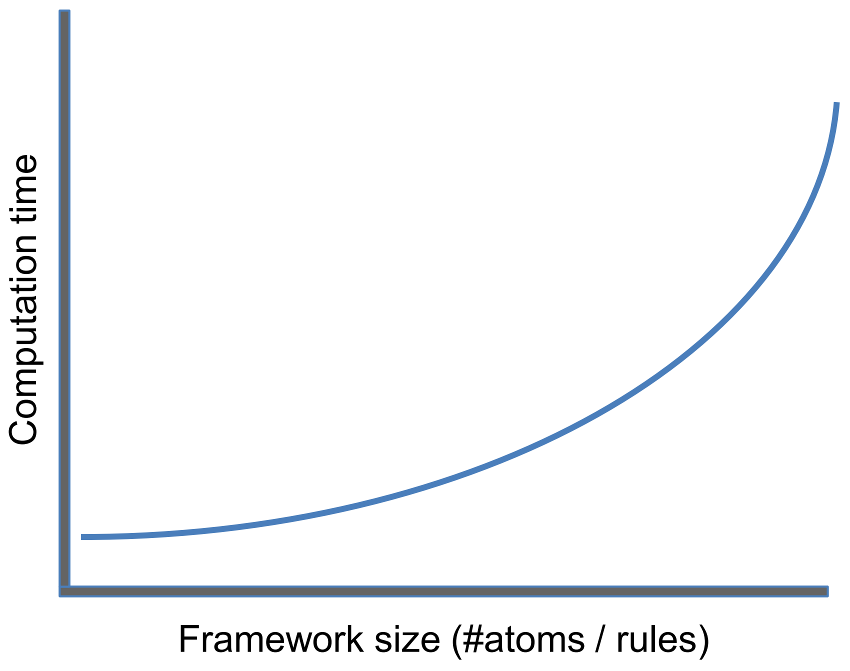
$\{b, c\}$ attacks $\{d\}$ (R3)

$\{a, b, d\}$ attacks $\{c\}$ (R1)

$\{a, b, d\}, \{b, c\}$

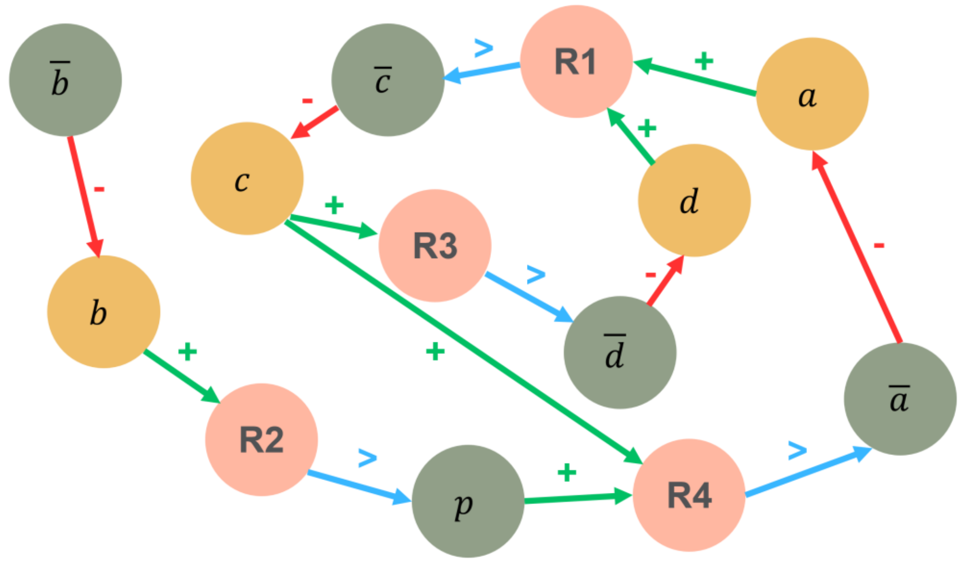
- Models defeasible reasoning explicitly
- Used in decision support, XAI, planning, causal discovery, etc.

The Challenge: Scalability



- Stable semantics → NP-complete
- Exact solvers time complexity grows exponentially
- Need scalable approximate methods as already available for AFs

Our Solution



- Encode ABA as a heterogeneous dependency graph
- Use GNNs to approximate credulous acceptance
- Use predictions to build stable extensions

Dependency Graph

We introduce a novel graph representation:

- Three types of nodes: assumption, claim and rule
- Three edge types:
  - Attack (-) edges
  - Support (+) edges
  - Derive (>) edges

ABAF D:

Assumptions A = $\{a, b, c, d\}$

Claims = $\{p, \overline{a}, \overline{b}, \overline{c}, \overline{d}\}$

Rules:

R1: $\overline{c} \leftarrow a, d$

R2: $\mathrm{p} \leftarrow \mathrm{b}$

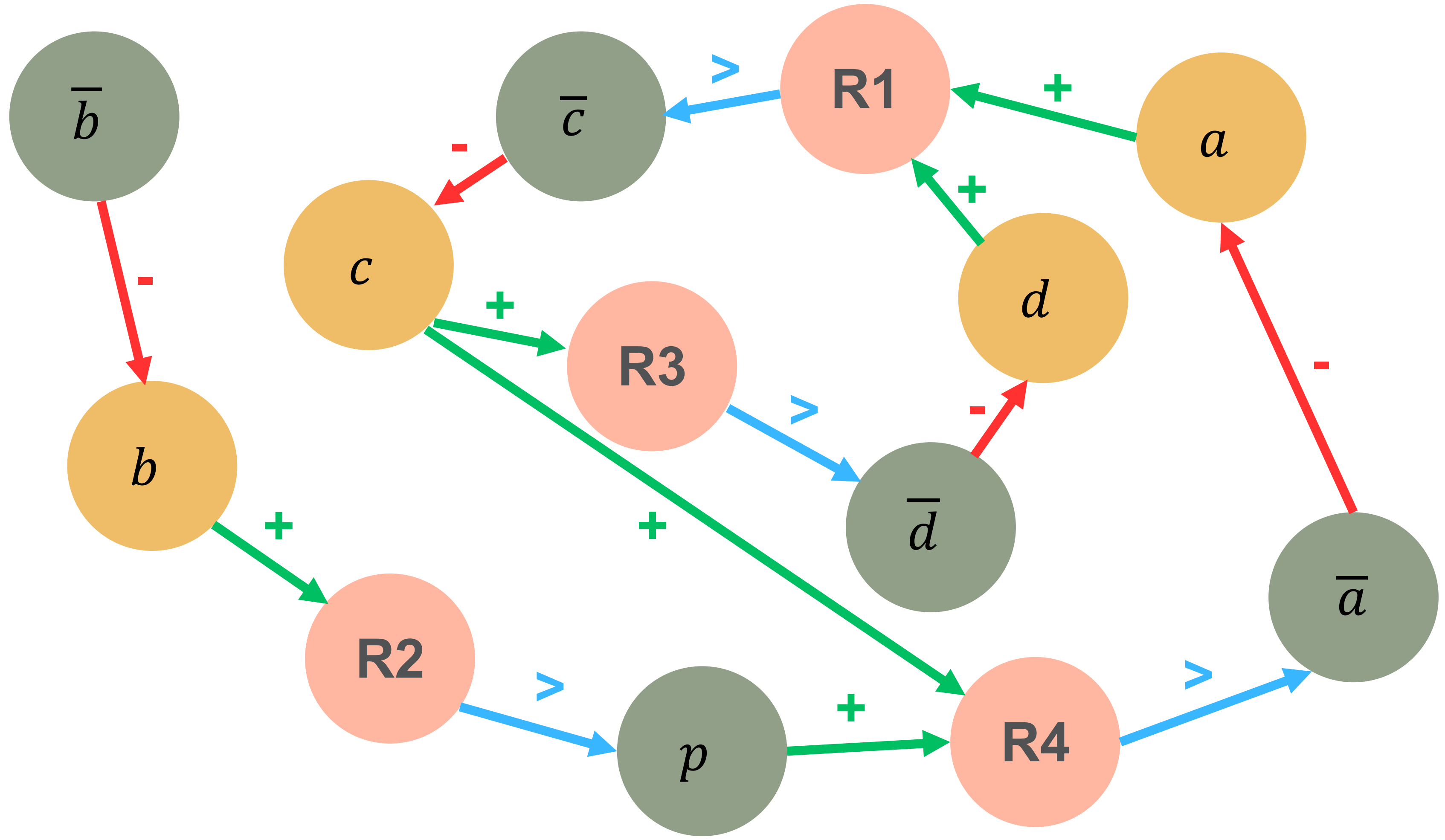
R3: $\overline{\mathrm{d}} \leftarrow \mathrm{c}$

R4: $\overline{\mathrm{a}} \leftarrow \mathrm{p}, \mathrm{c}$

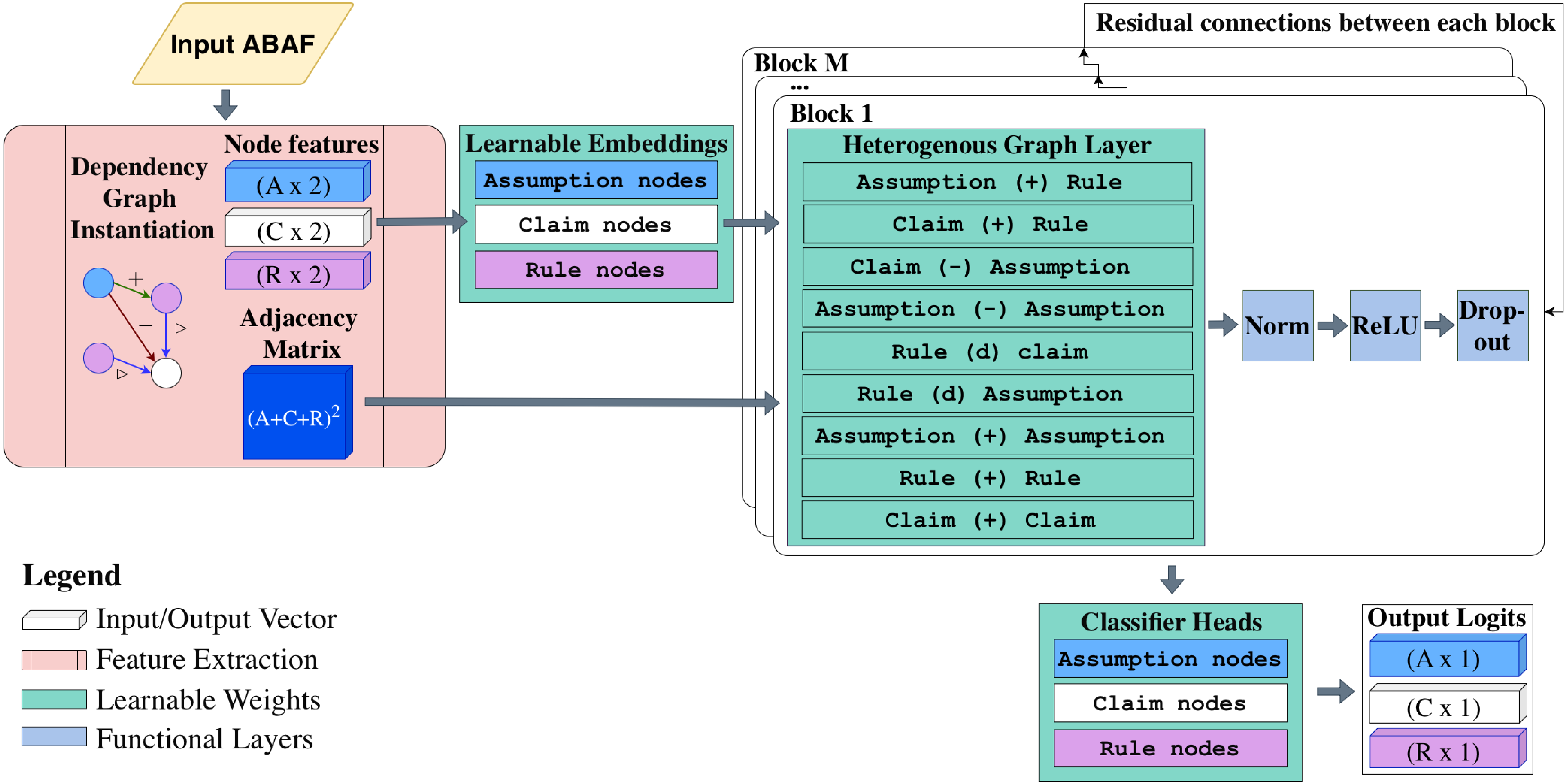
Contraries

$\{\overline{a}, \overline{b}, \overline{c}, \overline{d}\}$

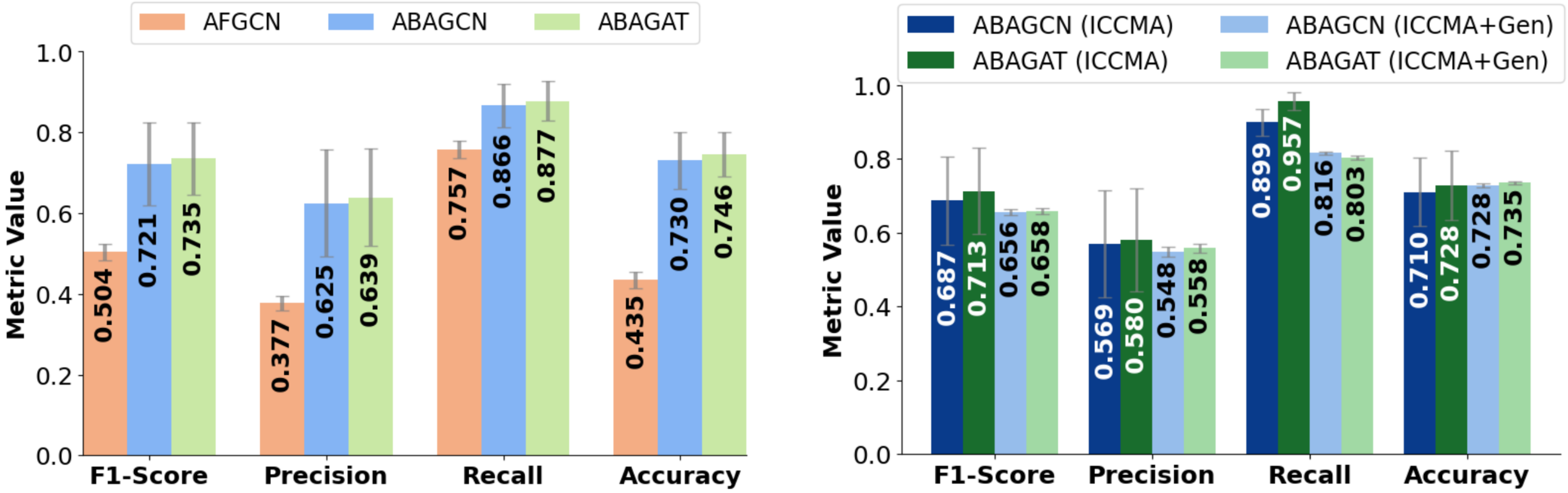
Dependency Graph:



Model Architecture



Experimental Results



(a) AF vs ABA GNNs on ICCMA data with $|\mathcal{L}| < 100$

(b) CGN vs GAT on ICCMA and generated data

Extension Calculation

Pass Dependency Graph of the ABAF into GNN

↓

Get probability predictions for all assumptions

↓

Select an assumption with the highest probability

↓

Remove accepted assumption, add to our extension set and update the ABAF

Repeat the steps until the ABAF contains no more assumptions

Conclusion & Future Work

Novel dependency graph representation: Faithful structural representation of ABAFs.

Develop two heterogeneous GNN architectures: ABAF → Dependency Graph → (trained) GNN

- **Handling non-flat ABAFs:** fine tune final model on non-flat ABAFs and update extension enumeration to handle them.
- **Improving models' performance:** increase diversity of training set, topological features, and experiment with different model architectures

Extension Calculation: Embed the final GNN model into an algorithm to build stable extensions

- **Augmenting extension calculation with exact methods:** Calculate the grounded extension, exact methods for ABAFs with no rules.
- **Improving the search process of the extension algorithm:** when the extension generated is invalid, backtrack and explore alternative paths (accepted assumptions)

Implementation & Testing: Extensive experiments and available GitHub with weights ready for use

Questions?

Check Out Our Repo

Read Full Paper

IMPERIAL